

Subiectul III (30 de puncte) - Varianta 040

Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Utilizăm metoda backtracking pentru generarea tuturor modalităților de a scrie numărul 6 ca sumă a cel puțin două numere naturale nenule. Termenii fiecărei sume sunt în ordine crescătoare. Soluțiile se generează în ordinea: 1+1+1+1+1+1, 1+1+1+1+2, 1+1+1+1+3, 1+1+1+4, 1+5, 2+2+2, 2+4 și 3+3. Se aplică exact aceeași metodă pentru scrierea lui 9. Câte soluții de forma 2+... vor fi generate? (6p.)

a. 2 b. 3 c. 4 d. 5

Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.

2. Pentru definiția alăturată a subprogramului `f`, stabiliți ce valoare are `f(23461)`. (4p.)
- ```
int f(int x)
{if(x<10){if(x%2!=0)return 0;
return x;
}
if(x%2!=0)
return f(x/10);
return f(x/10)+x%10; }
```

3. Pe prima linie a fișierului `bac.in` se află un număr natural nenul `n` ( $n \leq 1000$ ), iar pe a doua linie a fișierului se află un șir format din `n` numere naturale, despărțite prin câte un spațiu, fiecare număr fiind format din cel mult 4 cifre. Scrieți un program C/C++ care citește numerele din fișier și care afișează pe ecran mesajul **DA** dacă elementele pare în șir sunt în ordine crescătoare, iar cele impare sunt în ordine descrescătoare și mesajul **NU** în caz contrar.

**Exemplu:** dacă fișierul `bac.in` are conținutul

|   |    |      |    |     |    |      |     |    |
|---|----|------|----|-----|----|------|-----|----|
| 8 | 10 | 1133 | 12 | 331 | 42 | 1354 | 221 | 13 |
|---|----|------|----|-----|----|------|-----|----|

alăturat, pe ecran se va afișa: **DA** (10p.)

4. Considerăm definite subprogramele:
- `pr`, care primește prin intermediul parametrului `x` un număr natural cu cel mult 4 cifre și returnează 1 dacă numărul este prim și 0 în caz contrar;
  - `sdiv` care primește prin intermediul parametrului `y` un număr natural cu cel mult 4 cifre și returnează suma tuturor divizorilor numărului `y`.

**a)** Scrieți numai antetul subprogramelor `pr` și `sdiv`. (4p.)

**b)** Scrieți un program C/C++ care citește de la tastatură un număr natural `n` ( $n < 1000$ ) și care afișează pe ecran toate numerele mai mici decât `n` cu proprietatea că suma divizorilor lor este număr prim. Se vor utiliza apeluri utile ale subprogramelor `pr` și `sdiv`.

**Exemplu:** dacă `n=20`, atunci programul va afișa: 2 4 9 16. (6p.)